Advanced Scientific Tools and Systems (ASTS) Group

Group Lead: David Matusow, GSFC Code 588

david.matusow@gsfc.nasa.gov, 301-286-9231

February 1, 2001

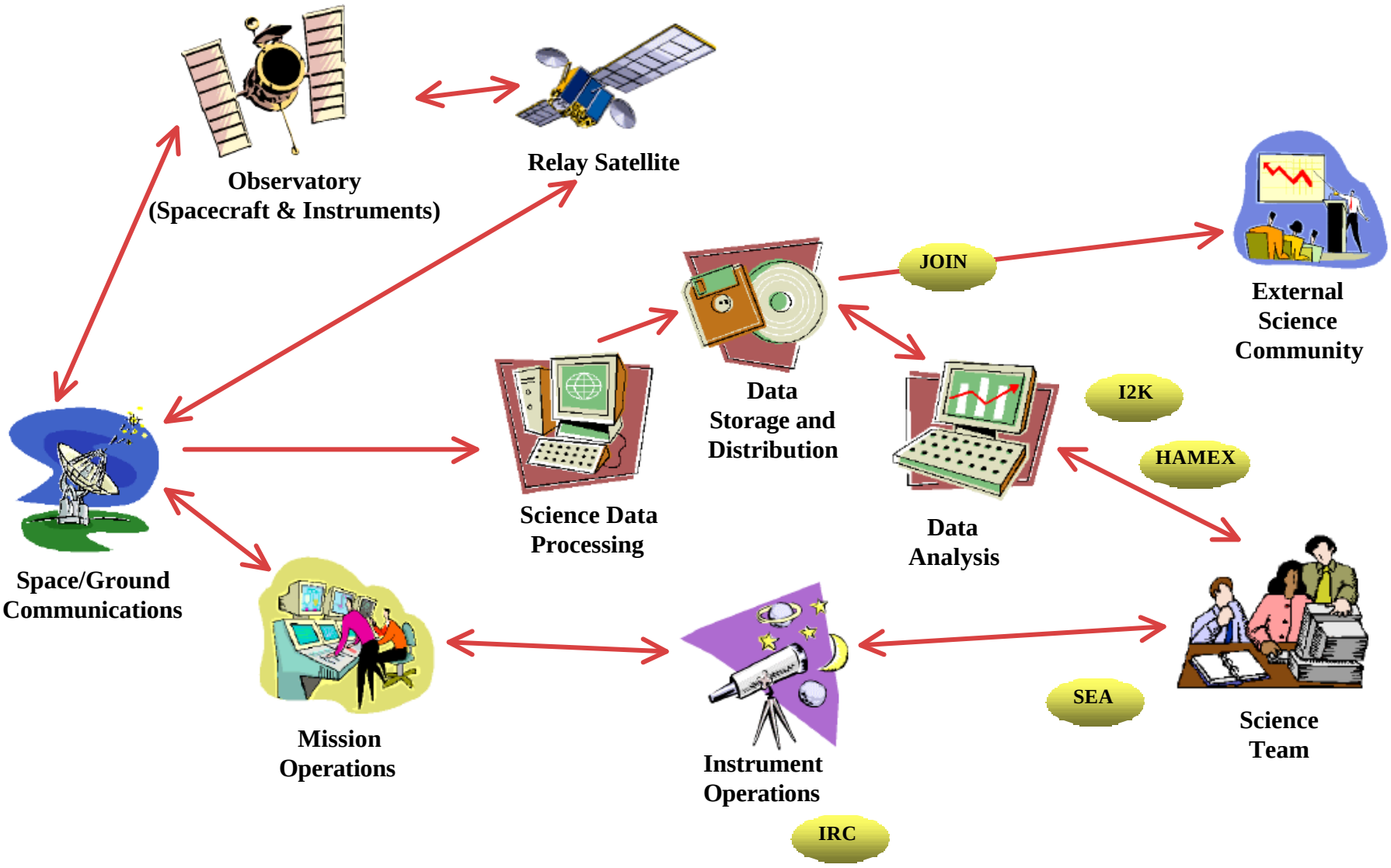
Agenda

- Group Purpose / Charter
- Projects:
  - Scientists Expert Assistant (SEA)
  - Image2000
  - Jini Object Information Network (JOIN)
  - Instrument Remote Control (IRC)
  - Handheld Mars Exploration (HAMEX)
- Discussion / Questions

ASTS Group Purpose

- Provide advanced tools and systems to scientists that foster an environment that enables science knowledge discovery through seamless and transparent access to information.
- Work with the scientific community to understand their needs and develop state-of-the-art information system solutions to solve those needs
- Research new technologies and methods for producing advanced software

Advanced Scientific Tools and Systems



Scientist's Expert Assistant (SEA)

SEA - Project Overview

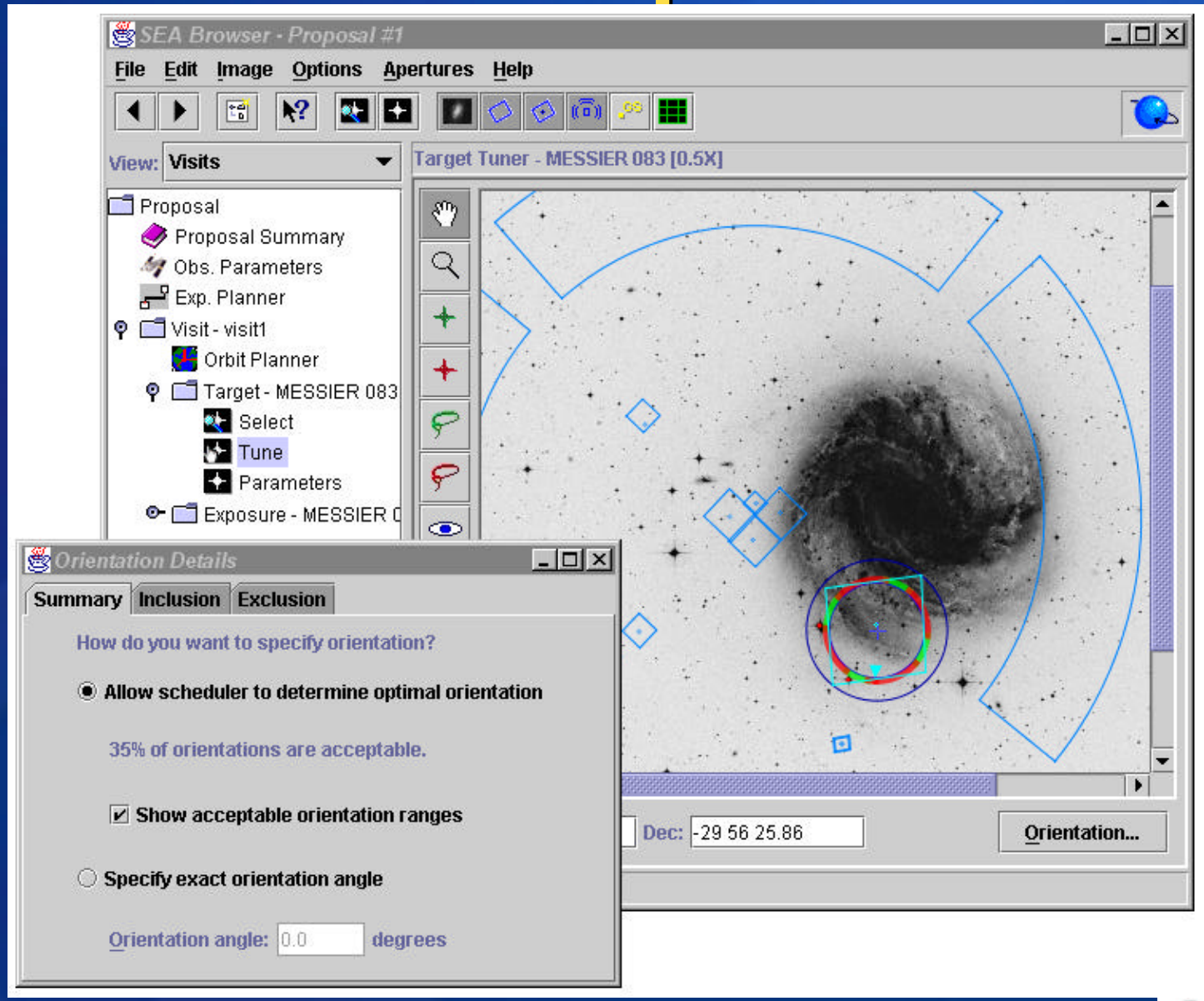
- Goal:
  - Research ways to reduce user support costs via a collection of integrated tools that assist an observer in specifying an observation
- SEA for NGST
  - Put the “Eye” back in observation using a variety of visualization tools
    - Visual Target Tuner
    - Intelligent, Rule Based Query Tool (Instrument and Dither Assistants)
    - Graphical Exposure Time Calculator, Orbit Planner
  - Adapted by Space Telescope Science Institute (STScI) [called the Astronomers Proposal Toolkit (APT)]
- SEA Simulation Facility (SSF)
  - Allows an observer to get an accurate prediction of the observation before committing resources
  - Breaks down an observation into elements of the light path to accomplish simulation

SEA - Technology

- Java
  - Provides platform independence*
  - Allows for smaller development time through use of add-on libraries (Java Advanced Imaging, etc.)
  - Object Oriented Language allows for greater code re-use and easier maintenance
- XML
  - Used for proposal storage
  - Language for configuration and preference files
  - Allows the use of the multitude of XML editors and parsers already written
  - Reduces the level of maintenance needed and increases the readability of the files

** = Requires Java 1.2, available on most versions of Unix, Windows and the Mac OS X beta*

SEA - Example Screen



SEA - Milestones / Schedule

- A) First Prototype - 05/01/1998
- B) Release 3 - 04/02/1999
- C) Release 4 - 10/30/1999
- D) Final Delivery to NGST - 03/01/2000
- E) Deliver SF prototype release - 09/2001
- F) SF release 2 (increased fidelity for instrument calibration) - 09/2002

SEA - Research Possibilities

- Investigate plug-In architectures to allow a generic way to adopt new observatories and instruments
- Investigate “GRID” possibilities for SEA to:
  - Share distributed data
  - Share compute power
    - Simulations might be very compute intensive
- Research new effective data simulation for the astronomical domain

Image2000

I2K - Project Overview

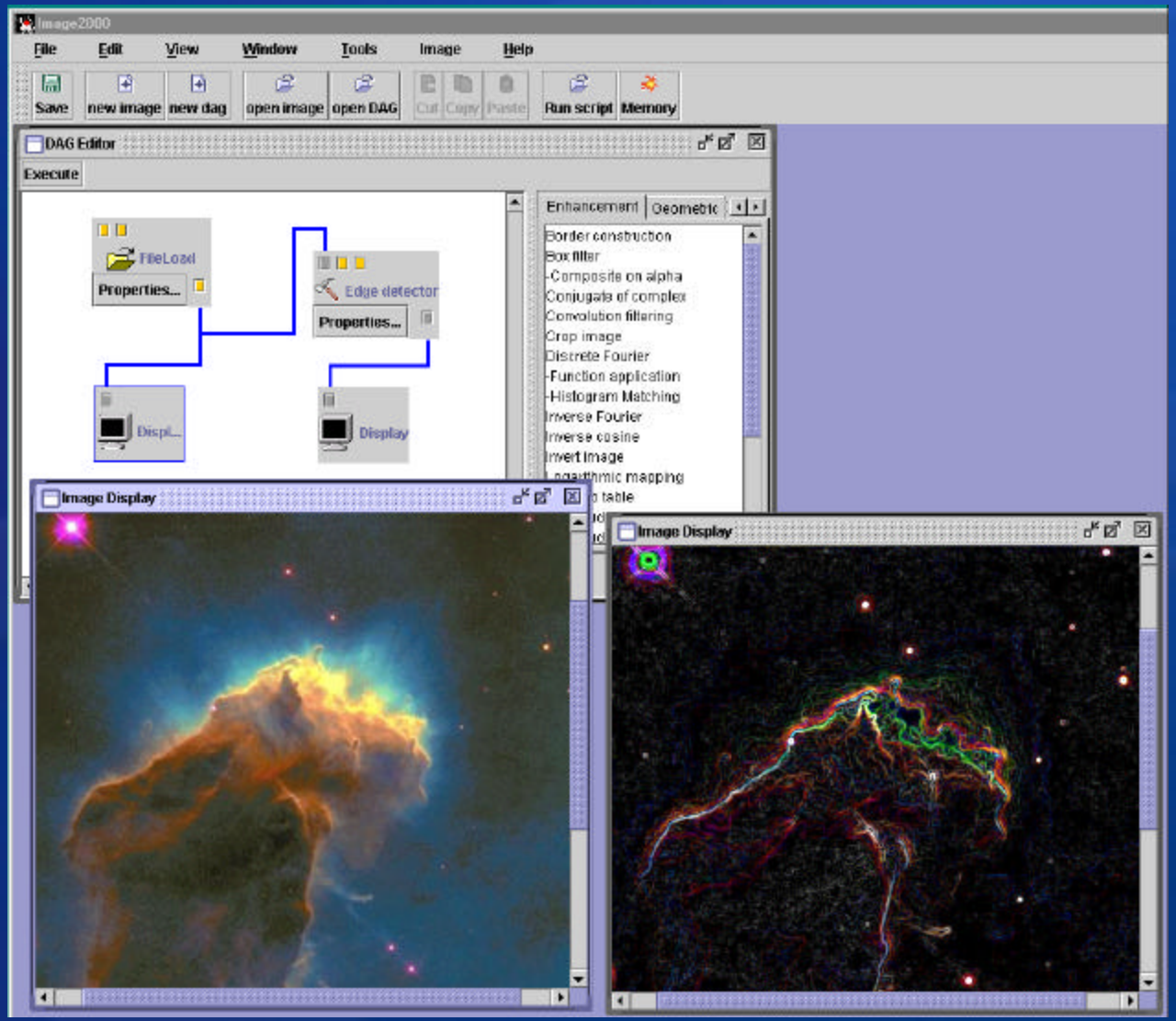
- **Goal:**
  - **Provide free Image processing tool for accessing and manipulating geo-referenced imagery, vector files, and remote sensing data**
  - **Used in education facilities to teach disciplines of remote sensing, biology, math and physics**
  - **Replacement for NIH Image tool, which only supported the Mac platform, and is no longer supported**
- **Allow easy customization of the interface for users of all levels**
- **Ability to add new image processing algorithms without writing new code**
- **Allow for the development of new image processing algorithms**
- **NOT exactly Photoshop® for Science Data**

I2K - Technology

- Java
  - Provides platform independence*
  - Allow for smaller development time through use of add-on libraries (Java Advanced Imaging, etc.)
    - Robust plug-in architecture extends I2K abilities through JAI components (Custom Java Beans, Java Script, and others)
  - Object Oriented Language allows for greater code re-use and easier maintenance
- Java Script
  - Scripting language - allows a user to add extended capabilities
- XML
  - User Interface entirely defined in XML for easy configuration
  - Allows the use of the multitude of XML editors and parsers already written
  - Reduces the level of maintenance needed and increases the readability of the files

** = Requires Java 1.2, available on most versions of Unix, Windows and the Mac OS X beta*

I2K - Example Screen



Advanced Architectures & Automation Branch

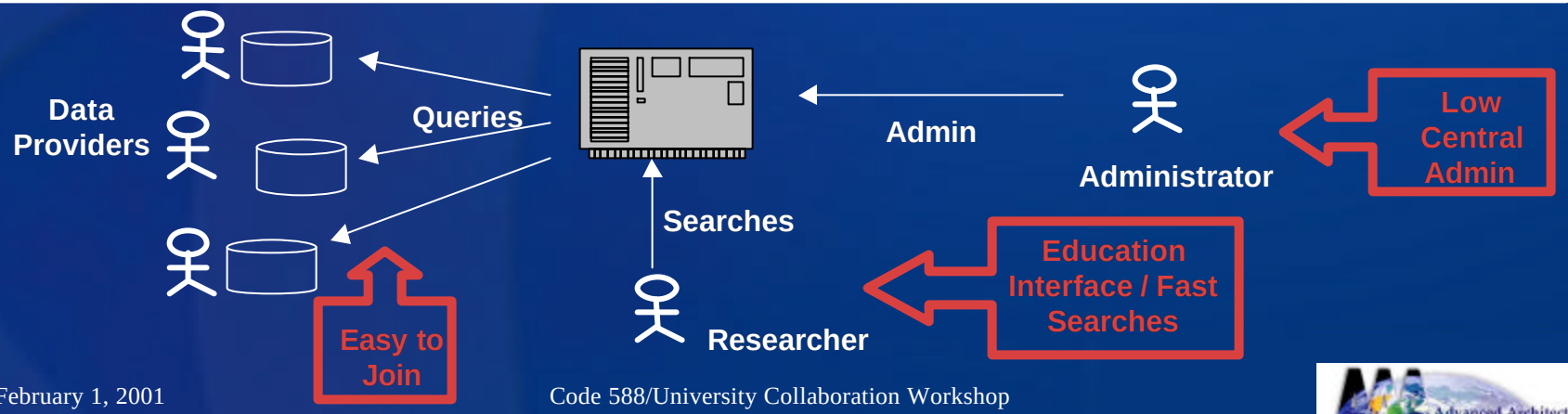
I2K - Milestones / Schedule

- A) First Prototype - 06/30/1999
- B) Final Phase 1 release - 11/30/1999
- C) Demo to Earth Science - 09/30/2000
- D) Phase 2 final release - 12/22/2000
- E) NASA evaluation - 06/30/2001
- F) Transfer to Univ of Va. for Maintenance - 7/31/2001

Jini Object Information Network (JOIN)

JOIN - Project Overview

- **Goal:**
  - **An ongoing prototyping effort investigating the use of Sun's Jini technology to facilitate efficient, decentralized, and distributed computing.**
- **JOINed Digital Library for Science Education is first project to implement JOIN.**
  - **Developed for GSFC's Earth Science Education and Outreach departments**
  - **Provide access to NASA's science education resources from archives distributed across the country**
  - **Not exactly "Napster" for Science Education Libraries**



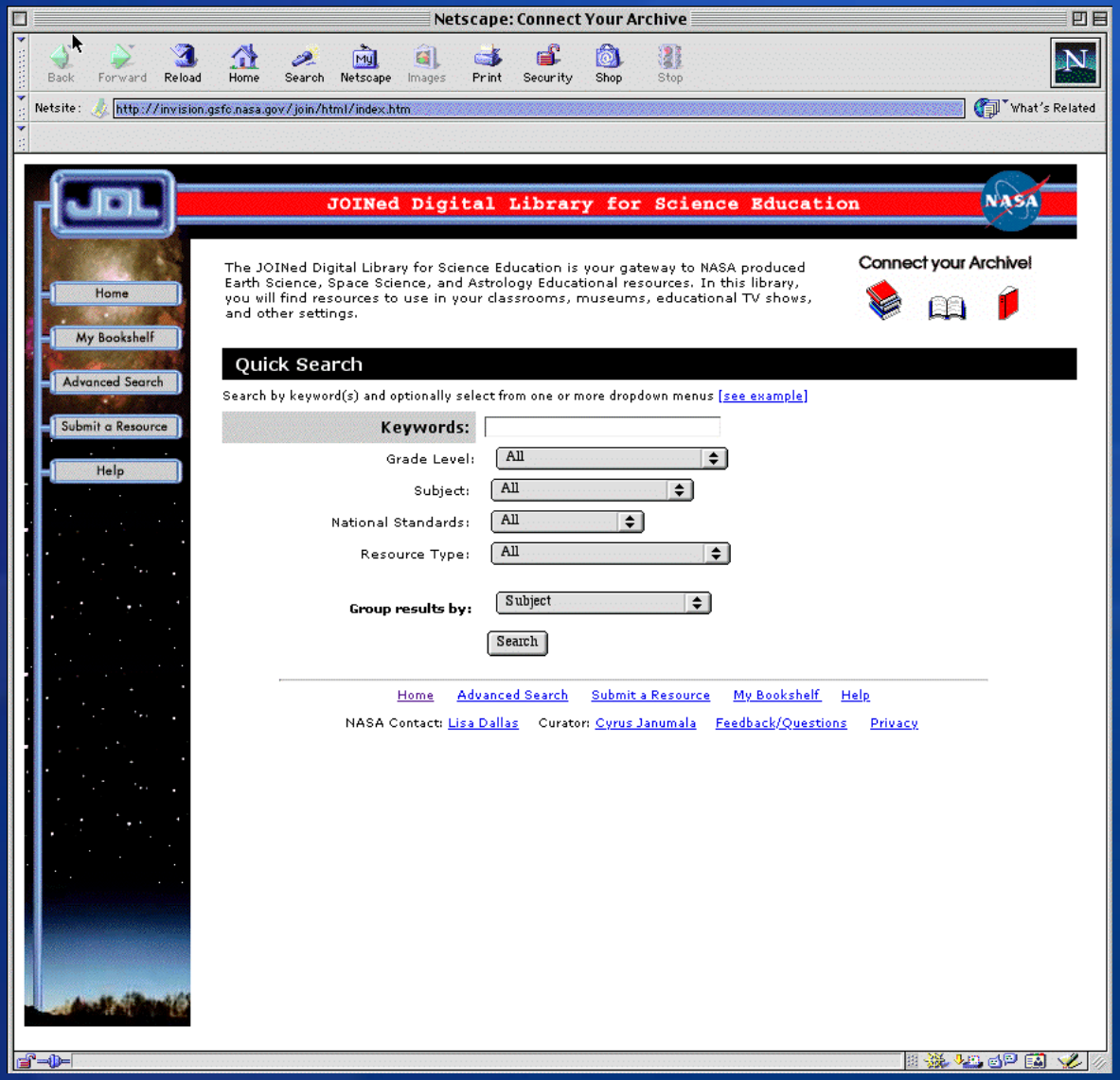
Advanced Architectures & Automation Branch

JOIN - Technology

- Java
  - Provides platform independence*
  - Allow for smaller development time through use of add-on libraries (Java Advanced Imaging, etc.)
  - Object Oriented Language allows for greater code re-use and easier maintenance
- XML
  - Storage of metadata files
- Digital library query protocol: SDLIP
- Jini
  - Interoperable framework with the following benefits:
    - 1) System is self-healing - resilient to network / machine failures
    - 2) Requires virtually no system administration
    - 3) Clients can easily locate and use available services
    - 4) Legacy services may also be accessed via a Jini wrapper

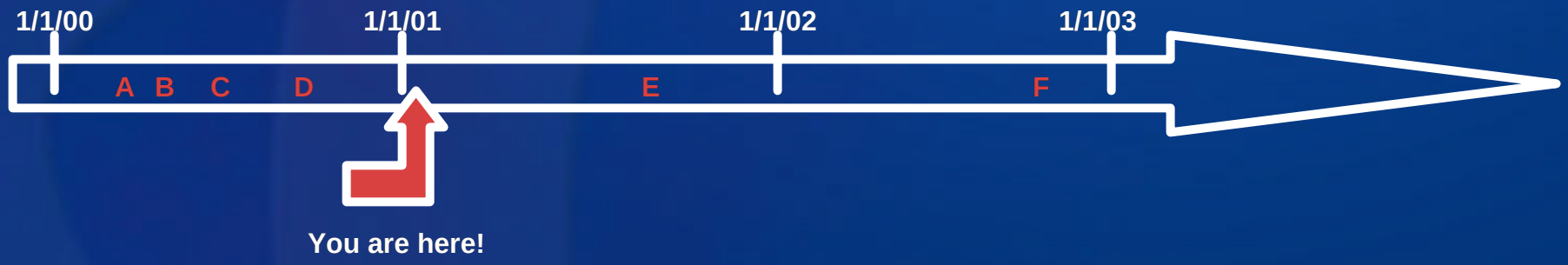
** = Requires Java 1.2, available on most versions of Unix, Windows and the Mac OS X beta*

JOIN - Example Screen



JOIN - Milestones / Schedule

- A) Requirements Review - 03/01/2000
- B) Design Review - 04/15/2000
- C) Selected Prototype Application - 06/15/2000
- D) Initial proof of concept - 08/15/2000
- E) Pilot demo and delivery - 09/2001
- F) Release 2 - 09/2002



JOIN - Research Possibilities

- Security for the JOINed Digital Library
  - How do we validate that content comes from an approved archive?
  - How do we allow a single researcher to securely add content to a database without having every file approved by a peer review committee?
    - Ex) Triana images will be automatically transmitted every 15 minutes. How can an archive verify that the images originated from the Triana server?

Instrument Remote Control (IRC)

IRC - Project Overview

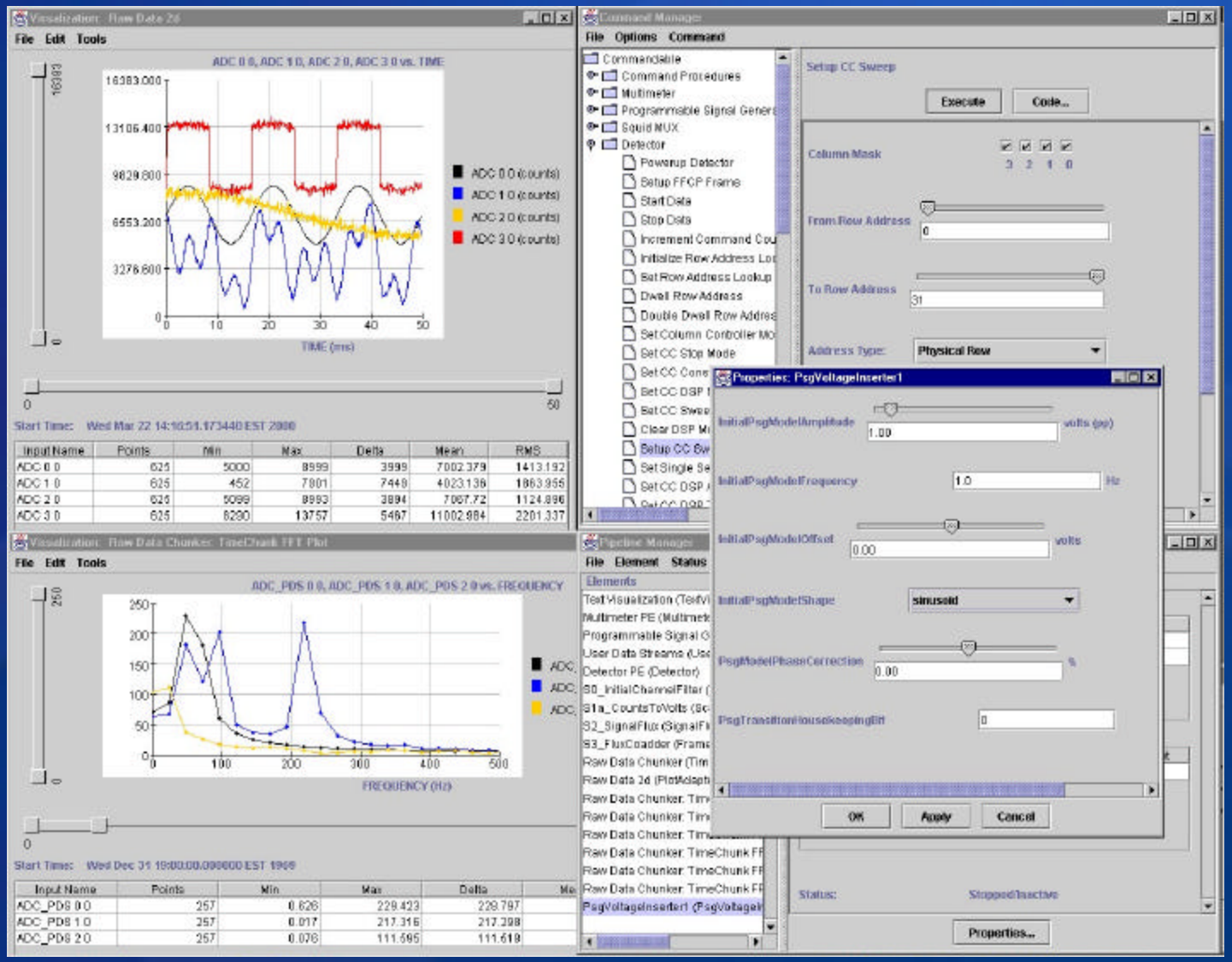
- **Goals:**
  - **Research a collaborative, adaptive framework for the distributed configuration, control, monitoring and data analysis of remote instruments**
  - **Provide an easy-to-use, intuitive Graphical User Interface that provides interactive or scripted real-time instrument control**
- **Current Objectives**
  - **Develop an extensible framework to which a wide variety of new instruments can be added with relative ease with reduced costs**

IRC - Technology

- Java
  - Provides platform independence*
  - Allow for smaller development time through use of add-on libraries (Java Advanced Imaging, etc.)
  - Object Oriented Language allows for greater code re-use and easier maintenance
- XML
  - Software driven by XML-based Instrument Description
    - GUI, command set, command formats
    - data pipeline algorithm descriptions
    - data streams (responses, images)
    - online help and documentation

** = Requires Java 1.2, available on most versions of Unix, Windows and the Mac OS X beta*

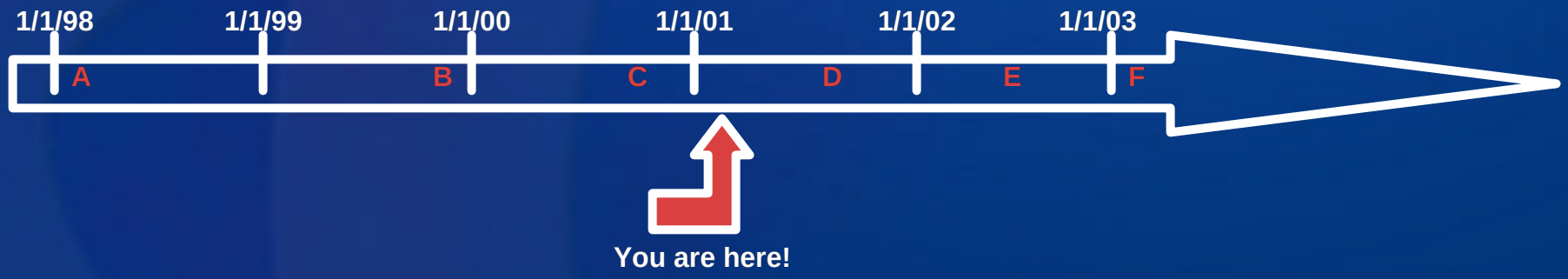
IRC - Example Screen



Advanced Architectures & Automation Branch

IRC - Milestones / Schedule

- A) South Pole Demo - 01/31/1998
- B) SPIRE final delivery- 12/31/1999
- C) Framework v3.0 - 10/30/2000
- D) Framework v4.0 (GUI customization) ~ 09/2001
- E) Framework v5.0 (Modeling Support) ~ 06/2002
- F) SOFIA mission support ~ 01/2003



IRC - Research Possibilities

- Techniques for modeling a subsystem or instrument behavior
  - Many techniques available, which is best?
- Adaptable GUI
  - Investigate ways to provide a dynamic, adaptable GUI so that it is highly configurable depending on the user:
    - Novice vs Expert user
    - Engineer vs. Scientist
- Fast storage and indexing techniques for data
  - Streaming data is currently stored using Java serialization

Handheld Mars Exploration (HAMEX)

Rovers on Mars (2003)

R/T Webcast
From Mars

JPL Web Server

GSFC HAMEX Website

Data Pull

Internet Requests and
Response

Web-enabled CellPhone

Laptop or PC with Palm 3C

Palm 7X with Web Service

Questions???